

## ارائه اپراتور جدید جایگزین پخش قطره جوهر در روش یادگیری فعال

سجاد حق زاد کلیدبری<sup>۱</sup>، دانشجوی دکتری؛ سعید باقری شورکی<sup>۲</sup>، استاد؛ ایمان اسمعیلی پایین افراکتی<sup>۳</sup>، استادیار

۱- دانشکده مهندسی برق- دانشگاه صنعتی شریف - تهران- ایران- sajjad\_haghzad@ee.sharif.edu

۲- دانشکده مهندسی برق- دانشگاه صنعتی شریف - تهران- ایران- bagheri-s@ee.sharif.edu

۳- دانشکده فنی و مهندسی - دانشگاه مازندران - بابلسر - ایران - i.esmaili.p@umz.ac.ir

**چکیده:** روش یادگیری فعال<sup>۱</sup> یکی از روش‌های یادگیری فازی است که الگوگرفته از پردازش در مغز انسان است. اپراتور پخش قطره جوهر، موتور اصلی پردازشی در این روش است که به دور از فرمول‌های پیچیده، به دنبال یافتن رابطه بین خروجی و هر یک از ورودی‌ها است. زیادبودن حافظه مورد نیاز برای پیاده‌سازی صفحات پخش قطره جوهر و همچنین حجم محاسبات زیاد لازم برای استخراج ویژگی‌ها از جمله مشکلات پیش روی اپراتور پخش قطره جوهر<sup>۲</sup> است. در این مقاله یک روش جایگزین برای اپراتور پخش قطره جوهر ارائه شده است که سبب کاهش چشم‌گیر پیچیدگی محاسباتی می‌شود. الگوریتم ارائه شده با استفاده از دو بردار حافظه به توصیف صفحات پخش جوهر می‌پردازد که مشکل اتلاف زیاد حافظه را حل می‌کند. الگوریتم پیشنهادی، ویژگی‌های مسیر باریک و پراکندگی داده در صفحات پخش قطره جوهر را که مهم‌ترین مفاهیم برای استفاده در مرحله استنتاج الگوریتم یادگیری فعال هستند را با صرف کمترین هزینه و زمان محاسباتی می‌یابد. برای بررسی صحت عملکرد الگوریتم، شبیه‌سازی‌هایی بر روی مجموعه داده‌های استاندارد در حوزه مدل‌سازی و طبقه‌بندی ارائه شده است. زمان و دقت الگوریتم پیشنهادی با روش یادگیری فعال، شبکه‌های عصبی چندلایه پرسپترون<sup>۳</sup> و سیستم استنتاج عصبی-فازی تطبیق‌پذیر<sup>۴</sup> مقایسه شده است.

**واژه‌های کلیدی:** شبیه‌سازی یادگیری مغز، روش یادگیری فعال، اپراتور پخش قطره جوهر، سیستم استنتاج فازی<sup>۵</sup>، طبقه‌بندی الگوها.

## A Novel Alternative Operator for Ink Drop Spread (IDS) in Active Learning Method (ALM)

S. Haghzad Klidbary, Ph.D. student<sup>1</sup>; S. Bagheri Shouraki, professor<sup>2</sup>; I. Esmaili Paeen Afrakoti, Assistant professor<sup>3</sup>

1- Faculty of Electrical Engineering, Sharif University of Technology, Tehran, Iran, sajjad\_haghzad@ee.sharif.edu

2- Faculty of Electrical Engineering, Sharif University of Technology, Tehran, Iran, bagheri-s@ee.sharif.edu

3- Departments of Engineering and Technology, University of Mazandaran, Babolsar, Iran, i.esmaili.p@umz.ac.ir

**Abstract:** Active Learning method is one of the fuzzy learning methods inspired from the computation of human brain. Ink drop spread operator is the main computational engine in ALM which without using any complex formula, seeks for the relationship between the output and the inputs of the system. One of the challenges of IDS operator is that not only a large memory is required for implementing IDS planes, but also extracting features imposes high computational costs. In this paper, one learning method, as a replacement for IDS operator, is represented that considerable reduces the computational complexity. The represented algorithm defines the IDS planes with only two memory vectors and solves the problem of huge wastage of memory in these planes. This algorithm starts to learn the available patterns in learning data to find two features of *Narrow Path* and *Spread* in planes which are the most important concepts in the level of active inference learning with minimum computational and time cost. To investigate the accuracy of algorithm's performance, some simulations in modelling and classification have been done on standard data sets. Time and accuracy of proposed algorithm is compared with traditional ALM, MLP and ANFIS algorithms.

**Keywords:** Brain learning simulation, active learning method, ink drop spread operator, fuzzy inference system, pattern classification.

تاریخ ارسال مقاله: ۱۳۹۶/۰۶/۲۸

تاریخ اصلاح مقاله: ۱۳۹۶/۰۸/۳۰

تاریخ پذیرش مقاله: ۱۳۹۶/۱۱/۰۳

نام نویسنده مسئول: ایمان اسمعیلی پایین افراکتی

نشانی نویسنده مسئول: ایران- بابلسر- بلوار شهید ذوالفقاری - بلوار دانشگاه - پردیس دانشگاه مازندران - دانشکده فنی مهندسی.

## ۱- مقدمه

تاکنون الگوریتم‌های محاسبات نرم بسیاری برای شبیه‌سازی عملکرد مغز انسان معرفی شده‌اند که می‌توان این تحقیقات را در دو دسته شبکه‌های عصبی مصنوعی و رویکرد منطق فازی دسته‌بندی نمود. الگوریتم‌های بیان شده در حل بسیاری از مسائل از جمله تخمین توابع، پیش‌بینی، داده‌کاوی و مسائلی نظیر کاربرد دارند. در منطق فازی برای مدل‌سازی فرآیندها برخلاف روش‌های کلاسیک، به‌جای بیان روابط پیچیده ریاضی، مسئله توسط یک سری متغیرهای زبانی و رابطه‌های "اگر و آنگاه" بیان می‌شود، که با این کار از محاسبات دقیق دوری شده و سبب می‌شود که عملکرد سیستم در برابر عدم قطعیت پایدار شود [۱]. پایداری و عملکرد مناسب روش‌های بهره‌گرفته از منطق فازی در کاربردهای مختلفی بیان شده است [۹-۲].

از جمله الگوریتم‌های مشهور در حوزه یادگیری ماشین (شبکه‌های عصبی و منطق فازی) می‌توان به MLP و ANFIS اشاره نمود. این الگوریتم‌ها در کاربردهای مختلفی نظیر مدل‌سازی، طبقه‌بندی و کاربردهایی نظیر آن استفاده می‌شوند. این الگوریتم‌ها به‌دلیل بهره‌گیری از سخت‌افزاری قابل قبول برای پیاده‌سازی [۱۲-۱۰] و همچنین دقتی خوب، جایگاهی مناسب در مقایسه با سایر الگوریتم‌های مرز دانش<sup>۶</sup> در یادگیری ماشین دارند [۱۰، ۱۳]. الگوریتم MLP که از قانون بازگشت خطا<sup>۷</sup> بهره می‌گیرد یکی از مشهورترین الگوریتم‌های یادگیری است و همچنین الگوریتم ANFIS که ترکیبی از شبکه‌های عصبی مصنوعی و منطق فازی است از جمله روش‌های مهم در حوزه نو فازی است.

در راستای پیاده‌سازی الگوریتم‌های محاسبات نرم یکی از مهم‌ترین مسائل، یافتن بستر مناسب سخت‌افزاری با حجم مناسب منابع برای پیاده‌سازی است. تقریباً تمامی روش‌های موجود از محاسبات ریاضی پیچیده بهره می‌گیرند که در محیط نرم‌افزاری مشکلی را ایجاد نخواهد نمود ولی در پیاده‌سازی سخت‌افزاری نیازمند منابع بسیار زیادی خواهند بود. ضمناً سرعت فرآیندهایی که در مغز آدمی رخ می‌دهد بسیار زیاد بوده و بنابراین باید راه‌حل‌ها و الگوریتم‌های سریع‌تر و با نگاه متفاوت یافت.

الگوریتم روش یادگیری فعال یکی از روش‌های مؤثر در حوزه محاسبات نرم است که در سال ۱۹۹۷ توسط شورکی ارائه شد [۱۴]. روش یادگیری فعال بر اساس نظام یادگیری و خبرگی مغز انسان که به‌دنبال یافتن و یادگیری یک ارتباط منطقی علت و معلولی در بین رخدادها است عمل می‌کند. این روش دارای ساختار ساده‌ای بوده و برخلاف الگوریتم‌های مشابه خود مانند روش‌های مدل‌سازی سوگنو-یاسوکاوا [۱۵] و روش تاکاگی-سوگنو [۵] از محاسبات پیچیده دور بوده و به‌دلیل نوع خاص فرآیند یادگیری، همگرایی آن بسیار سریع‌تر صورت می‌پذیرد [۱۶]. موراگامی و هوندا قابلیت بالای الگوریتم یادگیری فعال را در زمینه‌های محاسبات نرم نشان داده‌اند [۱۶]. مزایای زیاد الگوریتم یادگیری فعال نظیر سرعت بالا، مقاومت بالا در مقابل

نویز و عدم نیازمندی به فرآیندهای تکراری یادگیری، سبب کاربرد فراوان آن در دهه اخیر در حوزه‌های مختلفی نظیر کنترل و رباتیک [۱۹-۱۷]، مدل‌سازی [۲۵-۱۹] و شناسایی الگو [۲۶-۱۹] شده است.

ایده اصلی روش یادگیری فعال بر مبنای تقریب یک سیستم چند-ورودی-تک‌خروجی با استفاده از چندین زیرسیستم تک‌ورودی-تک-خروجی است که هر یک از این زیرسیستم‌ها (صفحات پخش قطره جوهر) بیانگر رابطه بین خروجی سیستم نسبت به هر یک از ورودی‌ها می‌باشند. از هر یک از صفحات پخش قطره جوهر دو ویژگی مسیر باریک (منحنی مشخصه) و پراکندگی داده استخراج می‌گردد که این ویژگی‌ها در مرحله استنتاج الگوریتم ALM مورد استفاده قرار می‌گیرند. افزایش این صفحات IDS با توجه به نیاز مسئله (افزایش تعداد متغیرهای ورودی و شکستن فضای متغیرهای ورودی)، رابطه مستقیم با پیچیدگی محاسباتی صفحات پخش قطرات جوهر و همچنین پیاده‌سازی سخت‌افزاری الگوریتم دارد. در صورت افزایش تعداد صفحات، حجم حافظه مورد نیاز الگوریتم به‌صورت چشم‌گیری افزایش خواهد یافت و همچنین زمان محاسباتی الگوریتم نیز زیاد خواهد شد. بنابراین یافتن بیانی جدید از صفحات پخش جوهر در کاهش بار محاسباتی و کاهش حجم حافظه مورد نیاز این الگوریتم تأثیرگذار خواهد بود و قابلیت سیستم را در محاسبات سیستم‌هایی با ورودی‌های زیاد و همچنین برخورد با سیستم‌های بلادرنگ افزایش می‌دهد.

در الگوریتم ارائه‌شده، با توصیفی متفاوت و یافتن توابع توصیف‌کننده صفحات پخش قطره جوهر، سبب حذف محاسبات ناشی از اپراتور پخش قطره جوهر (عدم نیازمندی به پخش قطرات جوهر سه‌بعدی) و همچنین کاهش تعداد سلول‌های حافظه مورد نیاز به دو سطر با رزولوشن  $R_{sn_x}$  شده است که در الگوریتم اصلی تعداد حافظه‌ها برابر با  $R_{sn_x} * R_{sn_x}$  است ( $R_{sn}$  سطوح کوانتیزاسیون محورهای مختصات است). در این الگوریتم یک بردار حافظه بیانگر رابطه ورودی خروجی خواهد بود و بردار دیگر درجه اعتقاد به رخداد خروجی مرتبط با هر داده ورودی را بیان می‌دارد. نحوه یادگیری الگوریتم به این صورت است که با آمدن هر داده آموزشی جدید، با استفاده از معیار فاصله بین داده‌های آموزشی و مقادیر ذخیره‌شده در این سطرها، مقادیر بردارهای حافظه با اعمال ضریبی از فاصله به‌روزرسانی می‌شوند. بنابراین از دلایل مناسب بودن و کارایی الگوریتم پیشنهادی می‌توان در درجه اول به کاهش حافظه مورد نیاز الگوریتم پیشنهادی اشاره نمود که با ارائه توصیفی از صفحات پخش قطره جوهر نیازمندی به حجم محاسبات زیاد برای یافتن ویژگی‌های مسیر باریک و پراکندگی داده را کاهش داده است. همچنین کاهش محاسبات و کاهش حافظه سبب خواهد شد که الگوریتم بتواند در برخورد با سیستم‌هایی با تعداد ورودی‌های زیاد همچنان عملکرد مناسبی داشته باشد و نیز در کاربردهای بلادرنگ مورد استفاده قرار گیرد (در الگوریتم رایج ALM، با افزایش تعداد صفحات IDS، در شبیه‌سازی صورت‌گرفته توسط نرم‌افزار متلب، الگوریتم را با کمبود RAM مواجه می‌نمود). در نهایت

## ۲-۱- اپراتور پخش قطره جوهر

هسته اصلی روش یادگیری فعال اپراتور پخش قطره جوهر است که به نوعی بر پایه درون‌یابی فازی و تکنیک برازش منحنی پیاده‌سازی شده است. روش یادگیری فعال با بیان کیفی و رفتاری سیستم سعی در تقلید و شبیه‌سازی رفتار مغز دارد. عدم اطمینان به رخدادهای اطراف با استفاده از یک عملگر تحت نام اپراتور پخش قطره جوهر بیان می‌شود. عملکرد اپراتور پخش قطره جوهر از این واقعیت سرچشمه می‌گیرد که تجربیات، ماهیتی پیوسته دارند و فضای الگوهای یادگیری فقط محدود به داده‌های تجربی نخواهد شد. بنابراین خواص یک نقطه منحصرأ متعلق به همان نقطه نبوده و همسایگی آن نقطه را نیز شامل می‌گردد و با فاصله‌گیری از نقطه تجربه، درجه اطمینان ما به هم خواصی نقاط اطراف تجربه نسبت به خاصیت نقطه مرکزی تجربه کاهش می‌یابد. همانند شکل (۱) هرکدام از زیرسیستم‌های تک‌ورودی تک-خروجی، در صورت کوانتیزه کردن فضای ورودی و خروجی، صفحات مشبکی از نقاط داده تصویر شده به‌ازای رنج مشخصی از سایر ورودی‌ها می‌باشند. اثر اپراتور پخش قطره جوهر روی مجموعه نقاط تجربه یا همان داده‌های آموزشی، همانند چکاندن یک قطره جوهر روی هر نقطه در صفحه است. انتشار اطلاعات این‌گونه است که بعد از افراز ورودی‌ها و تشکیل صفحات پخش قطره جوهر، هر نقطه ورودی در صفحه ورودی- خروجی  $(x_i - y)$  مختص خودش مانند یک قطره جوهر چکانده می‌شود.

برای مشاهده عملکرد اپراتور پخش قطره جوهر، داده آموزشی  $(x_1, x_2, y) = (5, 10, 6)$  در یک سیستم دو- ورودی تک- خروجی در نظر گرفته می‌شود. روش یادگیری فعال برای این سیستم، در ابتدای کار به تعداد متغیرهای ورودی، دو صفحه  $x_1 - y$  و  $x_2 - y$  را که خالی از هرگونه قطره جوهر می‌باشند در نظر می‌گیرد. سپس اقدام به پخش قطره جوهر متناسب با هر صفحه پخش قطره جوهر می‌نماید. همانند شکل (۲-الف) در صفحه مشبک و کوانتیزه شده  $x_1 - y$  داده  $(x_1, y) = (5, 6)$  گسترانده شده و در صفحه  $x_2 - y$  داده  $(x_2, y) = (10, 6)$  گسترانده می‌شود. برای سادگی نمایش تنها یک صفحه پخش قطره جوهر در شکل (۲-الف) نمایش داده شده است.

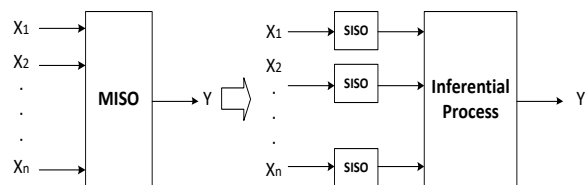
پس از گستراندن قطرات جوهر و همپوشانی قطرات، شکل‌های پیوسته‌ای در فضا همانند شکل (۲-ب) حاصل می‌گردد که پس از اتمام داده‌های آموزشی، واحد استخراج ویژگی در هر صفحه پخش قطره جوهر به استخراج دو ویژگی مسیر باریک (منحنی مشخصه) و پراکندگی داده می‌پردازد. این دو ویژگی بیانگر رابطه بین ورودی و خروجی می‌باشند (تداخل قطرات جوهر به‌نحوی، درون‌یابی فازی را مدل‌سازی می‌نماید).

الگوریتم پیشنهادی با کاهش محاسبات مرتبط با اپراتور پخش قطره جوهر و حذف محاسبات مرتبط با یافتن ویژگی‌های مسیر باریک و پراکندگی، سبب خواهد شد که در صورت ارائه ساختار سخت‌افزاری، کاهش چشم‌گیر تعداد المان‌های الکتریکی را در داشته باشد که همین امر کاهش سطح چپ‌پ و همچنین توان مصرفی آن نسبت به سخت‌افزارهای قبلی ارائه‌شده را نتیجه خواهد داشت (بنابراین ساده‌تر شدن پیچیدگی محاسباتی روش یادگیری فعال سبب خواهد شد که بتوان این الگوریتم را بر بسترهای سخت‌افزاری FPGA و ممریستوری<sup>۸</sup> به‌طور بهینه پیاده‌سازی نمود).

ساختار مقاله در ادامه به شرح زیر است: در بخش دوم مفاهیم روش یادگیری فعال و اپراتور پخش قطره جوهر ارائه شده است. الگوریتم ارائه‌شده در بخش سوم بررسی می‌شود. شبیه‌سازی‌ها و ارزیابی الگوریتم ارائه‌شده در بخش چهارم آورده شده است و کارایی الگوریتم در فرآیندهای مدل‌سازی و دسته‌بندی نسبت به الگوریتم‌های رایج پخش قطره جوهر و سایر الگوریتم‌های محاسباتی مورد مقایسه قرار گرفته است. در نهایت بخش پنجم شامل جمع‌بندی، نتیجه‌گیری و پیشنهادها برای کارهای آینده است.

## ۲-۲- روش یادگیری فعال

نحوه پردازش و انجام محاسبات در سیستم عصبی انسان ماهیتی کیفی و نادقیق داشته و همین امر نیازمندی به روش محاسباتی نرم (دارای عدم قطعیت) در شبیه‌سازی رفتار مغز را دوجندان می‌کند. روش یادگیری فعال نیز از روش‌های فازی بهره‌گرفته و بر این اصل استوار است که، انسان نگرشی غیردقیق به اتفاقات اطراف خود داشته و به‌جای ذخیره اعداد و ارقام، رفتار کلی سیستم را به‌صورت تصاویر نادقیق و مبهم یاد می‌گیرد (دوری از فرآیندهای پیچیده تکرارشونده ریاضی و عددی). انسان در برخورد با مسائل پیچیده سعی بر این دارد که با شکستن مسائل به مفاهیم ساده‌تر، و ایجاد یک ارتباط منطقی بین این مفاهیم، یک شناخت کلی از سیستم بیابد. روش یادگیری فعال با شکستن مسئله پیچیده به چندین مسئله ساده‌تر سبب درک-پذیری بالاتر مسئله می‌شود. عملکرد این روش در شکل (۱) نمایش داده شده است.



شکل ۱: در روش یادگیری فعال، یک سیستم پیچیده (MISO) به چند سیستم تک‌ورودی-تک‌خروجی (SISO) شکسته می‌شود.

با توجه به توضیحات بیان شده می‌توان روش یادگیری فعال را در دو بخش بررسی نمود: بخش اول را به‌روزرسانی صفحات پخش قطرات جوهر نام برد و بخش دوم را استنتاج نام برد، که در ادامه به معرفی این بخش‌ها خواهیم پرداخت.

شدت تاریکی هر نقطه یا شعاع پخش‌شدگی حول هر نقطه را در نظر گرفت. در ادامه روش استخراج ویژگی بیان شده در [۲۰] بررسی می‌شود. باید توجه داشت که به خاطر ذات نادقیق الگوریتم و اپراتور پخش قطره جوهر در نهایت جواب نهایی حاصل از روش‌های بیان شده، تغییر زیادی حاصل نمی‌گردد و انتخاب روش مورد استفاده وابسته به حجم سخت‌افزار و سرعت پردازش مورد نیاز است.

مجموعه‌ی  $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$  که شامل  $N$  داده آموزشی  $D$  بعدی به صورت  $x = (x_1, x_2, \dots, x_D)^T$  است را در نظر گرفته می‌شود. برای پیاده‌سازی الگوریتم ALM ابتدا باید رنج تغییرات متغیرهای ورودی و خروجی هر یک از صفحات پخش جوهر را کوانتیزه کنیم. در اینجا فرض بر این است که این سطوح کوانتیزاسیون برابر با  $R_{sn_x}$  و  $R_{sn_y}$  هستند.

با فرض اینکه  $p(x, y)$  یک نقطه در صفحه  $x_i - y$  باشد و تیرگی در این نقطه برابر با  $d(x, y)$  است، قطره جوهر را یک تابع محذب در نظر گرفته شده که مقدار آن در مرکز برابر با یک است. در این صورت به‌روزرسانی صفحه به‌ازای داده آموزشی  $(x_s, y_s)$  در ادامه آورده شده است:

$$p_{x,y} = \{p(x, y) \mid x \in X_i, y \in Y\} \quad (1)$$

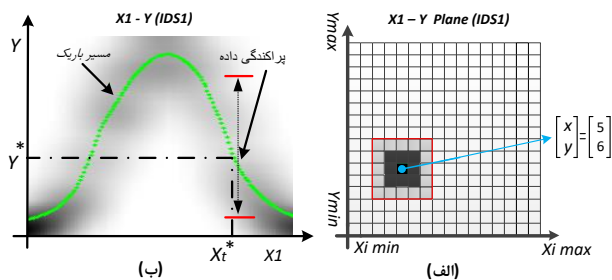
$$d(x_s + u, y_s + v) = d(x_s, y_s) + h(u, v), \quad -R < u, v < R \quad (2)$$

در رابطه فوق  $R$  بیانگر شعاع قطره جوهر و  $h$  شکل تابع قطره جوهر است. توابع  $\psi$  و  $\sigma$  به ترتیب به‌عنوان مسیر باریک و میزان پراکندگی در صفحه IDS را می‌توان به‌صورت زیر تعریف نمود:

$$\psi_{x_i}(x) = \{b \mid \sum_{y=y_{min}}^b d(x, y) \approx \sum_{y=b}^{y_{max}} d(x, y)\} \quad (3)$$

$$\sigma_{x_i}(x) = \max_{y \in Y} \{y \mid d(x, y) > T\} - \min_{y \in Y} \{y \mid d(x, y) > T\} \quad (4)$$

رابطه اول بیان می‌دارد که مقدار تابع مسیر باریک به‌ازای هر ورودی  $x$  برابر با  $b$  است به‌شرطی که مجموع جوهرهای ذخیره‌شده در سلول‌های بالای مختصات  $(x, b)$  در صفحه دوبعدی با مقداری تلورانس برابر با مجموع مقادیر جوهرهای ذخیره‌شده در سلول‌های پایین مختصات  $(x, b)$  باشد. رابطه دوم بیان می‌دارد که مقدار تابع پراکندگی در نقطه  $(x, b)$  متناسب با عرض مؤثر الگوی قرارگرفته بر روی صفحه IDS در ستون متناظر با  $x$  است و در این روابط پارامتر  $T$  بیانگر مقدار آستانه برای یافتن و تأثیرگذاری مقدار تیرگی ذخیره‌شده در سلول برای مقدار پراکندگی است. در شکل (۳) ساختار کلی روش یادگیری فعال برای یک سیستم با دو ورودی و یک خروجی نمایش داده شده است که البته در هر یک از ورودی‌ها دو افراز صورت گرفته شده است. همان‌طور که در شکل (۳) نیز مشاهده می‌شود تعداد واحدهای IDS



شکل ۲: عملکرد اپراتور IDS. (الف): ساختار یک واحد پخش‌کننده قطره جوهر (ب): الگوی فرضی ایجاد شده برای تعدادی داده آموزشی فرضی و استخراج مقادیر مسیر باریک و پراکندگی به‌ازای یک نقطه تست فرضی  $x_i^*$ .

توابع عضویت متعلق به هر قطره جوهر را می‌توان گاوسی، هرمی، مخروطی و یا هر تابع ۳ بعدی محذبی در نظر گرفت که در این توابع با فاصله‌گرفتن از مرکز تابع، درجه عضویت کاهش می‌یابد. در شکل (۲) قطرات جوهر با توابع عضویت هرمی در برخی نقاط با هم همپوشانی داشته و نواحی پیوسته‌ای با درجه تعلق‌های مختلفی را ایجاد نموده‌اند. در صورتی که لکه‌های جوهر همپوشانی داشته باشند، نقاط همپوشانی تیره‌تر و درجه اعتقاد ما نسبت به این نقاط فضا افزایش خواهد یافت.

## ۲-۲- استنتاج در روش یادگیری فعال

بعد از اعمال اپراتور پخش قطره جوهر بر روی صفحات دوبعدی، ویژگی‌های مسیر باریک و پراکندگی که در مرحله استنتاج الگوریتم ALM مورد استفاده قرار می‌گیرند، استخراج می‌شوند. مسیر باریک رفتار کلی خروجی سیستم نسبت به تغییرات ورودی را بیان می‌دارد و مقدار پراکندگی بیانگر مقدار اهمیت هر متغیر ورودی  $x_i$  در مقایسه با سایر متغیرهای ورودی در تعیین مقدار خروجی سیستم است. همان‌طور که در شکل (۲-ب) نیز مشاهده می‌شود، به‌ازای برخی از مقادیر ورودی، پراکندگی نقاط بیشتر و به‌ازای برخی نقاط دیگر پراکندگی کمتر است. اگر به‌ازای مقادیری از ورودی رنج تغییرات خروجی زیاد باشد نشان می‌دهد که خروجی وابستگی کمتری نسبت به آن متغیر دارد و تغییر ورودی‌های دیگر باعث تغییرات بیشتر خروجی می‌شود. بنابراین می‌توان با استفاده از دانش میزان پراکندگی نقاط، یک سیستم چندورودی و تک‌خروجی را با یک سیستم تک‌ورودی و تک‌خروجی و با یک خطای قابل‌قبول تقریب زد. در صفحات IDS ای که پراکندگی نقاط زیاد است، می‌توان با شکست و افراز رنج ورودی‌هایی که سبب افزایش پراکندگی شده‌اند، سبب افزایش دانش استخراج‌شده از اطلاعات موجود شد و با این عمل، می‌توان دقت را زیاد و خطا را کم نمود.

روش‌های مختلفی برای محاسبه مسیر باریک وجود دارد که می‌توان به عملگر استخراج ماکزیمم، عملگر استخراج میانگین و مقایسه مقادیر مجموع متوالی شدت تاریکی جوهر اشاره نمود [۱۶، ۲۹، ۳۰]. برای یافتن شعاع پخش حول هر نقطه مسیر باریک می‌توان

$$\beta_i = \frac{\Gamma_i * \frac{1}{\sigma_{x_i}(x_i)}}{\sum_{j=1}^c \Gamma_j * \frac{1}{\sigma_{x_j}(x_j)}} \quad (10)$$

با آمدن داده جدید، مقادیر مسیر باریک و پراکندگی در نقطه داده تست محاسبه شده و در نهایت بر مبنای این ویژگی‌ها، در بخش استنتاج خروجی بدست می‌آید. همان‌طور که در روابط دیده می‌شود استنتاج در روش ALM، جمع وزن‌داری از مقادیر مسیرهای باریک بدست آمده از صفحات IDS است. برای نقاطی از  $x$  های ورودی که سلول‌های متناظر با  $y$ ، خالی از قطرات جوهر باشند، مقادیر  $\psi_{x_i}(x)$  و  $S_{x_i}(x)$  برابر با  $\frac{y_{max}}{2}$  در نظر گرفته می‌شود.

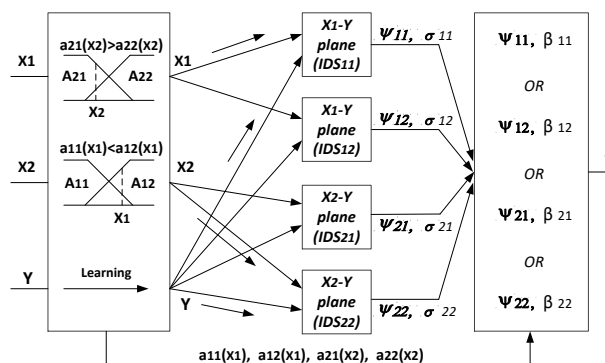
در الگوریتم ALM در صورت رسیدن به دانشی ناکافی و نادقیق، با شکستن سیستم به مجموعه‌ای از زیرسیستم‌های ساده‌تر درصدد کسب دانش بیشتر در آن زمینه خواهد بود و به این ترتیب هر یک از سیستم‌های تک‌ورودی تک‌خروجی نسبت به بخش معینی از سایر ورودی‌ها رفتارشان بررسی می‌شود (به این نکته باید توجه شود که تقسیم‌بندی بیش از حد دامنه متغیرهای ورودی در صورت کم بودن تعداد داده‌های آموزشی افزایش خطای خروجی را در پی خواهد داشت). از جمله مزایای روش یادگیری فعال این است که ترتیب ورود داده‌ها اهمیت ندارد، عمل به‌روزرسانی در این روش برخلاف الگوریتم‌های شبکه عصبی کاملاً محلی بوده و با آمدن داده جدید نیازی به به‌روزرسانی سرتاسری نیست، همچنین نیازمند پیکربندی اولیه نیست و با فرارگرفتن در محیط آزمایش به‌مرور با کسب تجربه و تعامل با محیط شروع به یادگیری می‌کند و در این الگوریتم ارائه یک‌بار داده‌های آموزشی به الگوریتم کفایت می‌کند. در شبکه‌های عصبی مصنوعی با توجه به مقادیر اولیه وزن‌ها، جواب‌های متفاوتی حاصل می‌شود و همچنین زمان آموزش‌ها متفاوت است، ولی در روش ALM همواره پایداری الگوریتم وجود دارد و مدل به پاسخ نهایی همگرا می‌شود.

روش یادگیری فعال با توجه به ذات نادقیق الگوریتم و نگرش نادقیق آن به داده، توانایی قابل‌قبولی را در مدل‌سازی و کنترل سیستم‌های نویزی از خود به نمایش گذاشته است [۲۸، ۳۰]. نکته‌ای که باید به آن توجه نمود این است که شعاع پخش قطره جوهر عامل بسیار مؤثر در عملکرد، کارایی، سرعت همگرایی و خطای خروجی است. تعیین شعاع پخش قطره جوهر رابطه مستقیم با تعداد داده‌های آموزشی، چگالی داده‌ها و رزولوشن صفحات دارد و مقدار آن معمولاً درصدی از رزولوشن، فشردگی و نحوه پراکندگی داده‌ها در فضا در نظر گرفته می‌شود. تعیین تعداد سطوح کوانتیزاسیون مرتبط با هریک از متغیرهای ورودی و خروجی وابسته به نوع مسئله، سرعت و دقت مدنظر کاربر خواهد بود.

مرتبط با هر متغیر ورودی با توجه به تقسیم‌بندی دامنه سایر متغیرهای است. رابطه زیر بیانگر تعداد صفحات IDS هر متغیر است:

$$l_i = \prod_{i=1, i \neq i}^N m_i \quad (5)$$

در رابطه فوق  $m_i$  تعداد تقسیم‌بندی دامنه ورودی به‌ازای هر متغیر است و با مجموع تمامی  $l_i$ ، تعداد صفحات IDS کل بدست خواهد آمد (به ازای  $N$  متغیر ورودی).



شکل ۳: ساختار کلی عملکرد روش یادگیری فعال در برخورد با یک سیستم شامل دو ورودی و یک خروجی به همراه دو افزاز روی هر یک از ورودی‌ها.

در ادامه قوانین مرتبط با مدل‌سازی سیستم در شکل (۳)، آورده شده است:

$$\begin{aligned} R_{11} &: \text{if } x_2 \text{ is } A_{21} \text{ (small) then } y \text{ is } \psi_{11} \\ R_{12} &: \text{if } x_2 \text{ is } A_{22} \text{ (big) then } y \text{ is } \psi_{12} \\ R_{21} &: \text{if } x_1 \text{ is } A_{11} \text{ (small) then } y \text{ is } \psi_{21} \\ R_{22} &: \text{if } x_1 \text{ is } A_{12} \text{ (big) then } y \text{ is } \psi_{22} \end{aligned} \quad (6)$$

در نهایت خروجی نهایی حاصل از جمع وزن‌دار قوانین فوق خواهد بود که برابر است با:

$$y \text{ is } \beta_{11}\psi_{11} \text{ or } \beta_{12}\psi_{12} \text{ or } \beta_{21}\psi_{21} \text{ or } \beta_{22}\psi_{22} \quad (7)$$

که  $\psi$  بیانگر مسیر باریک،  $\sigma$  بیانگر شعاع پراکندگی است و مقدار  $\beta$  از (۸) بدست می‌آید.

$$\beta_{ij} = \frac{\frac{1}{\sigma_{ij}} \Gamma_{ij}}{\frac{1}{\sigma_{11}} \Gamma_{11} + \frac{1}{\sigma_{12}} \Gamma_{12} + \frac{1}{\sigma_{21}} \Gamma_{21} + \frac{1}{\sigma_{22}} \Gamma_{22}} \quad (8)$$

در (۸) متغیر  $\Gamma_{ij}$  مقدار اعتقاد مرتبط با هر صفحه IDS است (درجه درستی هر قانون)، بنابراین در نهایت برای یک سیستمی که دارای  $C$  صفحه IDS است، خروجی از (۹) پیروی می‌کند.

$$y(x) = \sum_{i=1}^c \beta_i \psi_{x_i}(x_i) \quad (9)$$



### ۳- الگوریتم پیشنهادی (NALM)

واقعیتی که باید به آن توجه نمود این است که سرعت پردازش‌هایی که در مغز آدمی رخ می‌دهد بسیار بالا بوده و الگوریتم‌های کنونی جوابگوی نیازمندی ما نیستند. بنابراین باید به دنبال راه‌حل‌های سریع‌تر و جدیدتری یافت و یا اینکه نحوه نگاه خود را به مسائل پیرامون تغییر یابد. در روش یادگیری فعال، عمل استنتاج بر مبنای ویژگی‌های استخراج‌شده از واحدهای IDS است و بنابراین کارایی روش ALM تا حد بسیار زیادی وابسته به نحوه عملکرد و پیاده‌سازی واحدهای IDS است. در نتیجه از چالش‌های پیش‌روی سیستم محاسباتی IDS و روش یادگیری فعال، حجم زیاد محاسباتی و حافظه زیاد مورد نیاز الگوریتم برای نگهداری صفحات سیستم محاسباتی IDS و استخراج ویژگی‌های مسیر باریک و پراکندگی است. در سیستم محاسباتی IDS حافظه مورد نیاز برابر با کل فضای داده است. در [۳۱] روشی جایگزین برای کاهش حافظه و در نهایت پیاده‌سازی آسان و کم‌حجم سخت‌افزار مورد نیاز اپراتور IDS پیشنهاد شده است. که در آن هر دو نقطه همسایه اقلیدسی با میانگین مقادیر آن‌ها جایگزین می‌شود و در نتیجه مقدار حافظه مورد نیاز به میزان هفت برابر کاهش یافته است، ولی با این حال هنوز این الگوریتم توانایی عملکرد مناسب در پردازش‌های بلادرنگ را ندارد و بنابراین باید راهکاری جایگزین برای کاهش پیچیدگی محاسبات IDS یافت. بنابراین، تغییر پارادایم یادگیری در روش ALM سبب تسریع شبیه‌سازی نرم‌افزاری و همچنین بهبود راه‌حل‌های سخت‌افزاری این الگوریتم خواهد شد.

اولین قدم مؤثر در ارائه الگوریتم جایگزین الگوریتم IDS، ارائه یک نگاشت مناسب برای نمایش و بیان کل فضا است. نگاشت ارائه‌شده باید بر مشکلاتی نظیر پیچیدگی محاسباتی، بهره‌مندی از معادلات ساده و قابلیت پیاده‌سازی و رسیدن به حداکثر دقت ممکن غلبه کند و همچنین توانایی عملکرد مناسب در انواع کاربردهای موجود برای الگوریتم IDS و همچنین در کاربردهای بلادرنگ را داشته باشد. در الگوریتم ارائه‌شده با استفاده از دو بردار به توصیف صفحات IDS پرداخته شده است و تمامی اطلاعات مرتبط با صفحه IDS را در این بردارها ذخیره شده است. برحسب اینکه داده‌های آموزشی در صفحه IDS نسبت به این خطوط در چه فاصله‌ای قرار گرفته‌اند، جابجایی این خطوط با توجه به یک تابع گاوسی به سمت داده‌ها صورت خواهد گرفت و این دو بردار برای توصیف مسیر باریک و پراکندگی خواهد بود. همانند سایر الگوریتم‌های یادگیری، الگوریتم ارائه‌شده از دو فاز آموزش و تست تشکیل شده است که به ارائه آن‌ها پرداخته شده است.

#### ۳-۱- مقداردهی اولیه و الگوریتم یادگیری

مجموعه آموزشی  $S$ ، که شامل  $N$  داده آموزشی  $D$  بعدی فرض شده است. هر یک از بردارهای توصیفی صفحه IDS با  $v(x)$  نمایش داده می‌شود که  $v_{Narrow\_path}$  بیانگر تغییرات مسیر باریک در صفحه IDS است.  $v_{Spread}$  نشان‌دهنده حدود تغییرات خروجی سیستم حول مسیر

باریک بوده که همان پراکندگی است و بیانگر درجه اعتقاد به هر مقدار خروجی در صفحه IDS مرتبط خواهد بود. نحوه یادگیری الگوریتم این‌گونه است که با آمدن هر داده آموزشی جدید، مقادیر ذخیره‌شده در این آرایه‌های حافظه با توجه به فاصله‌هایشان نسبت به مقدار  $y$  داده آموزشی ورودی به‌روزرسانی می‌شود. همان‌طور که پیش‌تر بیان شد با انجام کوانتیزاسیون مقادیر  $y$  سیستم به محدوده صفر تا رزولوشن  $R_{sn_y}$  نگاشت می‌شود.

$$v_{Narrow\_Path}(initial) = R_{sn_y} / 2 \text{ و } v_{Spread}(initial) = R_{sn_y} / 2$$

مقادیر اولیه بردارهای حافظه در مرحله آموزش الگوریتم هستند (که البته این بردارها در روش پیشنهادی هر مقدار دلخواهی می‌توانند داشته باشند و در نتیجه خروجی تأثیری نخواهد داشت) و به‌ازای هر داده ورودی با سنجش فاصله این داده با بردارهای توصیفی هر صفحه IDS، اقدام به به‌روزرسانی بردارهای توصیفی می‌شود. اگر فرض شود  $q(x)$  نقطه‌ای روی محور  $x$  باشد و  $Q(x) = \{q(x) | x \in X_f\}$ ، با فرض داده آموزشی  $(y_s, x_s)$ ، به‌روزرسانی آرایه‌های حافظه به‌صورت معادلات زیر پیشنهاد می‌شود:

$$v_{Spread\_min}(x_s + u) = v_{Spread\_min}(x_s) + \quad (11)$$

$$\beta_1 \times g(u) \times [y_s - v_{Spread\_min}(x_s)]$$

$$v_{Spread\_max}(x_s + u) = v_{Spread\_max}(x_s) + \quad (12)$$

$$\beta_1 \times g(u) \times [y_{Spread\_max}(x_s) - y_s]$$

$$v_{Narrow\_Path}(x_s + u) = \frac{(v_{Spread\_max} + v_{Spread\_min})}{2} \quad (13)$$

که  $v_{Spread\_min}$  و  $v_{Spread\_max}$  حدود پایین و بالای قطره جوهر در

صفحات پخش قطره جوهر هستند،  $g(u) = e^{\frac{-u^2}{2\sigma^2}}$  تابع گاوسی و با انحراف معیار  $\sigma$  و  $-Ir \leq u \leq Ir$  که  $Ir$  شعاع نرم‌کنندگی در روش IDS جدید نامیده می‌شود و مقادیر  $2/3 \sigma \leq Ir \leq 4\sigma$  را به خود می‌گیرد. با فرض  $v_{Spread} = v_{Spread\_max} - v_{Spread\_min}$  که بیانگر پراکندگی داده است، می‌توان (۱۱)، (۱۲) و (۱۳) را به‌صورت (۱۴) و (۱۵) بیان نمود.

$$v_{Narrow\_Path}(x_s + u) = v_{Narrow\_Path}(x_s) + \quad (14)$$

$$\beta_2 \times g(u) \times [y_s - v_{Narrow\_Path}(x_s)]$$

$$v_{Spread}(x_s + u) = v_{Spread}(x_s) + \quad (15)$$

$$2 * \beta_1 \times g(u) \times [y_{Narrow\_path}(x_s) - y_s]$$

پارامترهایی که در این بردارهای توصیفی مورد استفاده قرار می‌گیرد عبارت هستند از ضرایب یادگیری  $\beta_1$ ،  $\beta_2$  و انحراف معیار  $\sigma$ . ضرایب یادگیری برای تعیین میزان تأثیرگذاری و همگرایی بردارهای توصیفی IDS و انحراف معیار برای تعیین تعداد همسایه‌های اثرگذار داده آموزشی ورودی مورد استفاده قرار می‌گیرد. مقدار انحراف معیار زمانی که تعداد داده‌های آموزشی کم باشد بزرگ (برای پوشش فضای بیشتر)

در بخش بعدی به ارائه شبیه‌سازی‌هایی در حوزه مدل‌سازی و طبقه‌بندی برای بررسی عملکرد و کارایی الگوریتم پیشنهادی در مقایسه با الگوریتم ALM مرسوم و همچنین الگوریتم‌های ANFIS و MLP پرداخته شده است.

#### ۴- طبقه‌بندی

در این بخش جهت ارزیابی الگوریتم ارائه‌شده شبیه‌سازی‌هایی به‌ازای محک‌های استاندارد مختلف صورت پذیرفته است. در ابتدا به کاربرد مدل‌سازی پرداخته شده است و عملکرد آن را در مدل‌سازی توابع بررسی می‌شود و سپس به ارزیابی الگوریتم در بحث طبقه‌بندی پرداخته می‌شود. تمامی شبیه‌سازی‌ها در برنامه متلب ۲۰۱۳ با بهره‌گیری از Toolbox های Neural Network و Fuzzy Logic این نرم‌افزار با یک کامپیوتر شخصی با پردازنده Core i5، 2.4 GHz و 4GB رم صورت گرفته شده است. در تمامی شبیه‌سازی‌ها تعداد سطوح کوانتیزاسیون هر یک از محورها برابر با ۲۵۶ در نظر گرفته شده است.

#### ۴-۱- تخمین توابع

از تخمین توابع در مسائل زیادی نظیر کنترل، پیش‌بینی و مدل‌سازی رفتار سیستم‌های پیچیده و نظیر آن‌ها استفاده می‌شود. در این بخش به بررسی عملکرد الگوریتم پیشنهادی در مدل‌سازی دو تابع دو-ورودی تک-خروجی و غیرخطی  $Y_1$  و  $Y_2$  پرداخته می‌شود.

$$Y_1(x_1, x_2) = (1 + x_1^{-2} + x_2^{-1.5})^2, 1 \leq x_1 \leq 10, 1 \leq x_2 \leq 10 \quad (16)$$

$$Y_2(x_1, x_2) = (x_1 - 6 * \sin x_2)^2, 1 \leq x_1 \leq 10, 1 \leq x_2 \leq 10 \quad (17)$$

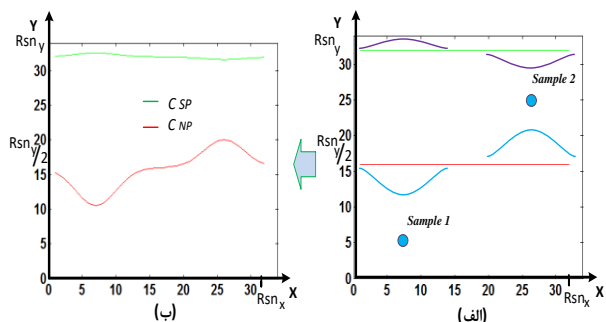
برای بررسی دقت الگوریتم و خطای مدل‌سازی از معیارهای  $FVU^{11}$  و همبستگی<sup>۱۲</sup> بیان‌شده در [۱۶] بهره گرفته و نحوه محاسبه آن‌ها با توجه به روابط زیر است:

$$FVU = \frac{\sum_{i=1}^k (y(x_i) - \hat{y}(x_i))^2}{\sum_{i=1}^k (y(x_i) - \bar{y})^2}, \quad \bar{y} = \frac{1}{k} \sum_{i=1}^k y(x_i) \quad (18)$$

$$Corr(z, t) = \frac{\sum_{i=1}^k (z_i - \bar{z}) \times (t_i - \bar{t})}{\sqrt{\sum_{i=1}^k (z_i - \bar{z})^2 \times (t_i - \bar{t})^2}} \quad (19)$$

در (۱۸)  $\hat{y}$  خروجی حاصل از مدل و  $k$  بیانگر تعداد داده‌های تست و در (۱۹)،  $t$  و  $z$  بیانگر میانگین دو بردار مورد مقایسه است.  $FVU$  هر چه کمتر باشد بیانگر افزایش دقت مدل و معیار همبستگی هر چه به یک نزدیک‌تر باشد بیانگر دقت بیشتر خروجی خواهد بود. در شکل (۵) و (۶) نمایش توابع  $Y_1$  و  $Y_2$  و تخمین آن‌ها آورده شده است.

و زمانی که تعداد داده‌های آموزشی زیاد باشد کوچک انتخاب می‌شود. مقادیر پارامترهای بیان‌شده در عملکرد و سرعت همگرایی الگوریتم بسیار مؤثر بوده و با توجه به نوع مسئله، تعداد داده‌های آموزشی و سرعت همگرایی مورد نیاز تعیین می‌گردد. زمانی که تعداد داده‌های آموزشی کم باشند می‌توان مقدار  $\beta$  و  $\sigma$  را برای افزایش دقت خروجی افزایش داد و همچنین این امکان برای الگوریتم پیشنهادی وجود دارد که همانند شبکه‌های عصبی مصنوعی می‌توان داده‌ها را چندین بار به الگوریتم نشان داد و سبب افزایش دقت شد که البته در اکثر مواقع مورد نیاز نیست. این پارامترها را می‌توان با سعی و خطا یا بهره‌گیری از الگوریتم‌های بهینه‌سازی نظیر الگوریتم ژنتیک بدست آورد. به‌عنوان مثالی برای بیان عملکرد و نحوه همگرایی الگوریتم، در شکل (۴) دو داده آموزشی فرضی به سیستم داده شده و نحوه به‌روزرسانی بردارهای توصیفی آورده شده است.



شکل ۴: الگوریتم پیشنهادی به‌ازای یک سیستم تک‌ورودی تک-

خروجی (الف): نتیجه اعمال دو داده فرضی  $(x_{q1}, y_{q1}) = (7, 5)$  و

(ب):  $(x_{q2}, y_{q2}) = (26, 24)$  بروزرسانی بردارهای مرتبط با صفحه IDS

به‌ازای مقادیر پارامترها  $\beta_1 = 0.1$ ،  $\beta_2 = 0.4$ ،  $Ir = 8$  و  $\sigma = 4$ .

الگوریتم ارائه‌شده، با ارائه توصیفی از صفحات IDS، ذخیره الگوی داده‌ها در فضای تجربه را با استفاده از دو آرایه حافظه مجزا انجام می‌دهد. برخلاف الگوریتم رایج IDS که برای ذخیره‌سازی فضای تجربه نیازمند بهره‌گیری از حافظه ماتریسی است در این الگوریتم با کاهش تعداد سلول‌های حافظه IDS به دو سطر با طول  $R_{snx}$ ، کاهش چشم‌گیری در حافظه مصرفی رخ داده است (کاهش تعداد سلول‌های حافظه از  $R_{snx} * R_{snx} * 2$  به  $2 * R_{snx}$ ). در نتیجه عمل پخش قطره جوهر حذف شده و پیچیدگی محاسباتی این اپراتور پخش در محاسبات وارد نمی‌گردد. بعد از فاز یادگیری در فاز تست، تنها کافی است به‌ازای هر داده ورودی تست، تنها مقادیر مرتبط از هر دو بردار حافظه خوانده شود و الگوریتم نیازمند انجام محاسبات مرتبط با یافتن ویژگی‌های هر صفحه IDS نیست. بنابراین می‌توان بیان داشت که، استفاده از بردارهای حافظه، سبب شده است که پردازش‌های محلی<sup>۱</sup> (پردازش‌هایی از نوع حافظه) جایگزین حافظه‌های حذف‌شده در روش IDS شدند و با به بیان دیگر، به‌صورت آنلاین حافظه ساخته<sup>۱</sup> می‌شود.

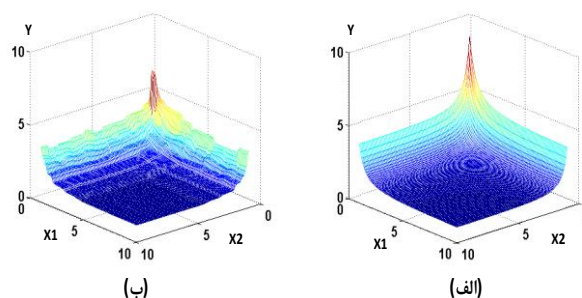
جدول به‌ازای محاسبه میانگین ۲۰ بار اجرای الگوریتم‌ها است. تعداد داده‌های آموزشی برابر با ۱۰۰۰، مقادیر پارامترها برای الگوریتم پیشنهادی و ALM همان مقادیر بیان‌شده در شبیه‌سازی شکل‌های (۵) و (۶) هستند.

جدول ۱: نتایج مرتبط برای الگوریتم پیشنهادی، الگوریتم‌های رایج ANFIS، MLP و ALM بر مبنای مقایسه معیارهای Corr. و FVU در برخورد با داده‌های نویزی آورده شده است (p=5).

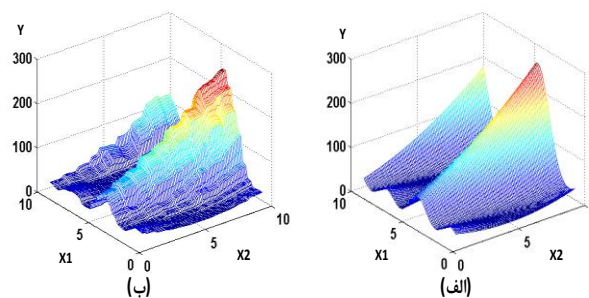
الگوریتم	تابع	FVU	Corr.	زمان (S.)
NALM	$Y_1$	$0.037 \pm 0.0091$	$0.9822 \pm 0.0081$	0.0903
	$Y_2$	$0.034 \pm 0.0086$	$0.9839 \pm 0.0084$	0.0852
ALM	$Y_1$	$0.036 \pm 0.0079$	$0.9875 \pm 0.0069$	0.2796
	$Y_2$	$0.038 \pm 0.0094$	$0.9821 \pm 0.0078$	0.2751
ANFIS	$Y_1$	$0.0094 \pm 0.0014$	$0.9937 \pm 0.0044$	1.1628
	$Y_2$	$0.0082 \pm 0.0021$	$0.9948 \pm 0.0029$	1.1923
MLP	$Y_1$	$0.0074 \pm 0.0012$	$0.9956 \pm 0.0033$	4.0112
	$Y_2$	$0.0061 \pm 0.0025$	$0.9964 \pm 0.0018$	4.7598

سرعت روش ALM بدلیل نداشتن فرآیندهای تکرارشونده در پروسه آموزش به‌میزان قابل‌توجهی سریع‌تر از روش ANFIS بوده [۳۲] و همان‌طور که در جدول نیز مشاهده می‌شود، سرعت روش پیشنهادی بیشتر از ALM بوده و بنابراین به‌مراتب خیلی سریع‌تر از روش‌های MLP و ANFIS است. نتایج جدول بیان می‌دارد الگوریتم ارائه‌شده توانسته مدل‌سازی بسیار خوبی از توابع را بیان دارد که نشان از عملکرد مناسب الگوریتم است. الگوریتم‌های MLP و ANFIS از مشهورترین روش‌ها با کارایی مناسب در مقابل نویز است [۳۳]. اپراتور پخش قطره جوهر در روش یادگیری فعال همانند یک نویز گاوسی بوده که در فرآیند آموزش به داده‌ها افزوده می‌شود و همین امر سبب افزایش قدرت یادگیری فعال همانند سایر روش‌ها در مقابله با داده‌های نویزی می‌گردد، که الگوریتم ارائه‌شده نیز از این موضوع بهره می‌برد.

در روش ALM، آموزش بصورت گروهی<sup>۱۵</sup> است یعنی همه‌ی داده‌های آموزشی با هم باید به الگوریتم ارائه شوند ولی در روش پیشنهادی بصورت نمونه به نمونه<sup>۱۶</sup> است و همین تفاوت روش آموزش از جمله مهم‌ترین دلایل سرعت بیشتر و حافظه کمتر الگوریتم پیشنهادی است. در هر دو روش آموزشی، همگرایی رخ می‌دهد ولی لزوماً هم‌ارز نخواهند بود، اما به‌دلیل اینکه فضایی محاسباتی دارای عدم قطعیت بوده، این خطا و عدم هم‌ارزی تأثیر چندانی در خروجی نهایی نخواهد داشت و در این فضای محاسباتی نمی‌توان بیان نمود که خروجی بهتر یا بدتر می‌گردد (البته متوسط همگرایی الگوریتم‌های Batch-Mode بیشتر از Sample-Mode می‌باشند). برای آنالیز پیچیدگی محاسباتی<sup>۱۷</sup> الگوریتم پیشنهادی، پیچیدگی زمانی و حافظه مورد نیاز الگوریتم



شکل ۵: مدل‌سازی تابع (الف): نمایشی از تابع  $Y_1$  بیان‌شده در (۱۶)، (ب): نمایش مدل‌سازی الگوریتم پیشنهادی.



شکل ۶: مدل‌سازی تابع (الف): نمایشی از تابع  $Y_2$  بیان‌شده در (۱۷)، (ب): نمایش مدل‌سازی الگوریتم پیشنهادی.

تابع  $Y_1$  به‌ازای ۱۰۰۰ داده آموزشی و مقادیر پارامترهای  $\sigma = 0.04$ ،  $\beta_1 = 0.74$ ،  $\beta_2 = 0.57$  و  $Ir = 15$  که در نهایت مقدار  $0.0212$ ،  $FVU = 0.9907$  و  $Corr = 0.9907$  (تعداد افزایندها برای هر دو متغیر برابر با ۱۰ است). تابع  $Y_2$  به‌ازای ۱۰۰۰ داده آموزشی و مقادیر پارامترهای  $\sigma = 0.05$ ،  $\beta_1 = 0.88$ ،  $\beta_2 = 0.7$  و  $Ir = 15$  که در نهایت مقدار  $0.0217$  و  $FVU = 0.9901$  است (تعداد افزایندها برای متغیر اول برابر با ۸ و برای متغیر دوم ۱۲ است). برای بررسی بیشتر، عملکرد الگوریتم‌ها در برخورد با داده‌های نویزی به مقایسه گذاشته شده‌اند که نتایج در جدول (۱) آورده شده است. درصد نویز افزوده‌شده به داده‌های آموزشی با رابطه زیر است:

$$y_{noisy} = y + \tilde{n}_p, \quad \tilde{n}_p = \frac{y \cdot \tilde{n} \cdot p}{100} \quad (20)$$

در (۲۰)،  $p$  بیانگر درصد نویز و  $\tilde{n}$  یک عدد تصادفی بین ۱ و ۱۰- بوده و  $Y$  داده‌های آموزشی هستند که در واقع نویز اعمالی در این مسئله، جایجایی تصادفی خروجی واقعی بر اساس درصدی از خود داده است. در شبیه‌سازی‌ها، شبکه MLP تعداد ۲ نرون به‌عنوان ورودی و ۱ نرون به‌عنوان نرون خروجی و در لایه پنهان ۱۰ نرون دارد. تابع فعالیت نرون‌ها از نوع سیگموئید و الگوریتم یادگیری لوبنبرگ-مارکار<sup>۱۳</sup> با نرخ یادگیری برابر با ۰/۰۹ در نظر گرفته شده است. در شبکه ANFIS، از تابع genfis1، با دو تابع عضویت دوزنقه‌ای برای هر متغیر استفاده شده است (تعداد ایپاک‌ها<sup>۱۴</sup> برای هر دو الگوریتم برابر با ۲۰۰ است). نتایج



بهره گرفته شده برای شبیه سازی آورده شده است. روش ارزیابی 10-Fold در تمامی شبیه سازی ها استفاده شده است. این ست داده ها و ست داده های مشابه در مقالات متعددی توسط الگوریتم های MLP و ANFIS و همچنین سایر الگوریتم های یادگیری ماشین شبیه سازی شده اند [۳۴، ۳۵] در این مقاله اقدام به شبیه سازی این الگوریتم ها بر پایه معیارهای ارزیابی یکسان صورت گرفته است که مقایسه منصفانه ای بین الگوریتم ها صورت گرفته شود.

جدول ۲: مشخصات ست داده های استفاده شده برای ارزیابی الگوریتم پیشنهادی.

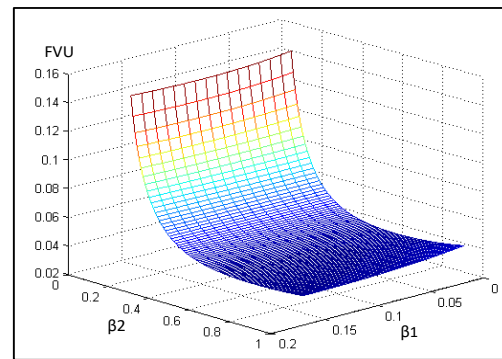
منبع	ست داده	داده	ویژگی	کلاس
http://sci2s.ugr.es/keel/category.php?cat=clas#sub2	Banana	۵۳۰۰	۲	۲
http://cs.joensuu.fi/sipu/datasets	3-Spiral	۳۱۲	۲	۳
http://cs.joensuu.fi/sipu/datasets	Aggregation	۷۸۸	۲	۷
http://sci2s.ugr.es/keel/category.php?cat=clas#sub2	Iris	۱۵۰	۴	۳
http://sci2s.ugr.es/keel/category.php?cat=clas#sub2	Wisconsin	۶۸۳	۹	۲

به عنوان اولین ست داده، Iris یک معیار استاندارد برای ارزیابی طبقه بندی بوده که شامل ۱۵۰ داده نمونه برداری شده از طول و عرض کاسبرگ و همچنین طول و عرض گلبرگ (در مجموع ۴ ویژگی) در سه نوع زنبق است. در این سه دسته همانند آنچه در شکل (۸) مشخص است، دو دسته از آن ها به صورت خطی قابل تمایز نیستند<sup>۱۸</sup>.

داده های ورودی در ابتدای کار نرمال استاندارد شده اند. در این مسئله، مدل به صورت سیستمی با چهار ورودی و یک خروجی در نظر گرفته شده که در ابتدا چهار صفحه پخش قطره جوهر وجود دارد و با توجه به دقت مورد نیاز این صفحات شکسته شده و بر تعداد صفحات پخش قطره جوهر افزوده خواهد شد. برای شبیه سازی ها در شبکه MLP تعداد ۴ نرون به عنوان ورودی و ۱ نرون به عنوان نرون خروجی و در لایه نهان ۷ نرون قرار گرفته شده است. تابع فعالیت نرون ها از نوع سیگموید در نظر گرفته شده است و الگوریتم یادگیری نیز لونیگ-مارکار که مقدار Rate یادگیری برابر با ۰/۰۳ در نظر گرفته شده است. در شبکه ANFIS، از تابع genfis2 استفاده شده است. تعداد ایپاک ها برای MLP برابر با ۱۰۰ و برای ANFIS برابر با ۴۰۰ است. در جدول (۳) به مقایسه خروجی حاصل از الگوریتم های مختلف پرداخته شده است. در روش پیشنهادی مقادیر پارامترهای  $\beta_1 = 0.009$ ،  $\beta_2 = 0.85$ ،  $\sigma = 7$  و  $Ir = 14$  (۵٪ رزولوشن صفحات) انتخاب شده اند، در روش ALM شعاع پخش قطره جوهر برابر با ۱۴ (۵٪ رزولوشن صفحات) است. تعداد افزایشها برابر برای هر دو الگوریتم و برای هر متغیر ۳ است.

پیشنهادی محاسبه شده است. پیچیدگی حافظه مورد نیاز در روش ALM برابر  $S(n) = O(R_{sn}^2)$  است و در روش پیشنهادی برابر با  $S(n) = O(R_{sn})$  است. پیچیدگی زمانی در روش ALM برابر با  $T(n) = O(n \times R^2 + R_{sn}^2)$  است و در روش پیشنهادی برابر با  $T(n) = O(n \times R + R_{sn})$  است.  $R$  و  $Ir$  بترتیب شعاع جوهر و شعاع نرم کنندگی می باشند و  $n$  بیانگر تعداد داده های آموزشی و  $R_{sn}$  همان رزولوشن سطوح است. بنابراین روش پیشنهادی در هر دو مورد بهتر عمل نموده و دلیل اصلی آن بهره گیری از دو بردار حافظه برای ذخیره سازی اطلاعات است.

با توجه به پیچیدگی های بدست آمده، می توان بیان داشت که پیچیدگی محاسباتی الگوریتم ارائه شده کمتر از الگوریتم پیشین است. در مقایسه ای دیگر به بررسی اثرات ناشی از تغییر دو پارامتر یادگیری  $\beta_1$  و  $\beta_2$  پرداخته شده و نتیجه تأثیر این پارامترها بر روی معیار FVU در شکل (۷) آورده شده است.

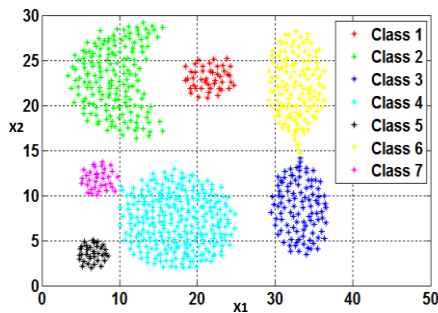


شکل ۷: اثر پارامترهای یادگیری بر روی دقت مدل سازی.

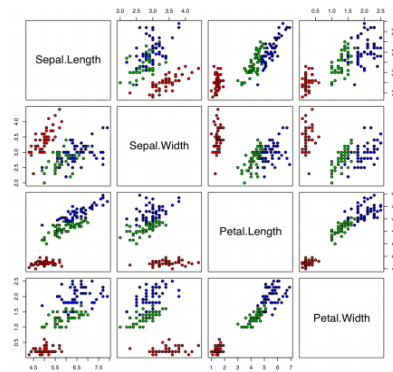
در شبیه سازی های انجام شده در شکل (۷) تعداد داده های آموزشی برابر با ۱۰۰۰ در نظر گرفته شده است و تعداد افزایشها برای هر دو متغیر یکسان بوده و برابر با ۱۲ است. همان طور که در این شکل مشاهده می شود، منحنی نزولی بوده و به ازای بازه ای از تغییرات پارامترهای همگرایی، خطای مدل سازی کمتر می شود. مقادیر پارامترهای یادگیری توسط کاربر و با توجه به تعداد داده های آموزشی انتخاب می شوند. در ادامه به بررسی نتایج الگوریتم پیشنهادی در مبحث طبقه بندی پرداخت خواهد شد و کارایی الگوریتم را در این نوع کاربرد بررسی می شود.

#### ۴-۲- دسته بندی

بدون شک یکی از قابلیت های مغز آدمی دسته بندی اشیای پیرامون خود است که این دسته بندی بر اساس ارتباط بین ویژگی های آن اشیا صورت می پذیرد. در گامی دیگر برای بررسی بیشتر و مقایسه کارایی الگوریتم پیشنهادی عملکرد الگوریتم در مسائل طبقه بندی بررسی شده است. در جدول (۲) مشخصات و ویژگی های انواع ست داده های



شکل ۹: ست داده Aggregation شامل هفت کلاس و ۷۸۸ داده.



شکل ۸: نمایش داده‌های ست داده گل زنبق، که در سه دسته زنبق وحشی تحت نام‌های Setosa (رنگ قرمز)، Versicolor (رنگ سبز) و Virginica (رنگ آبی) دسته‌بندی می‌شوند.

جدول ۴: میانگین دقت دسته‌بندی در ست داده Aggregation به ازای ۲۰ اجرای الگوریتم‌ها.

الگوریتم‌ها	دقت	زمان (S.)
NALM	99.22 ± 0.94	0.0323
ALM	99.50 ± 0.65	0.2651
ANFIS	99.49 ± 3.03	7.8148
MLP	99.10 ± 3.42	2.1687

برای مقایسه بیشتر و بررسی کارایی الگوریتم پیشنهادی به بررسی نتایج طبقه‌بندی بر روی ست داده‌های مختلف پرداخته شده است و نتایج در جدول (۵) آورده شده است.

جدول ۵: مقایسه الگوریتم پیشنهادی با روش‌های الگوریتم رایج

ALM, ANFIS, MLP به ازای ست داده‌های مختلف (دقت: Acc, Res:

کوانتیزاسیون، شعاع پخش قطره جوهر: R، شعاع نرم‌کنندگی:  $r$ ،  $\beta_1$  و  $\beta_2$  ضرایب یادگیری، تعداد توابع عضویت: MFs، تعداد ایپاک‌ها: Ep، Top ساختار شبکه، نرخ یادگیری: Lr).

الگوریتم	Wisconsin	3-Spiral	Banana	Acc.
ALM	97.37 ± 1.51	99.67 ± 0.37	89.88 ± 1.89	Acc.
	1,1,1,1,1,4,1,1,1	10,10	10,10	Par.
	128 × 128	128 × 128	160 × 160	Res.
NALM	12	18	16	R.
	96.77 ± 1.33	99.67 ± 1.02	89.78 ± 2.65	Acc.
	1,1,1,1,1,4,1,1,1	10,10	10,10	Par.
ANFIS	160 × 160	100 × 100	150 × 150	Res.
	12	24	14	Ir.
	0.03	0.05	0.01	$\beta_1$
ANFIS	0.74	0.96	0.39	$\beta_2$
	89.80 ± 0.65	99.69 ± 0.99	73.21 ± 2.06	Acc.
	genfis2/-	genfis3/-	genfis1/3	MFs.
MLP	300	400	800	Ep.
	94.44 ± 1.24	97.81 ± 4.18	74.34 ± 2.73	Acc.
	9-30-2	2-18-3	2-18-2	Top.
MLP	200	500	200	EP.
	0.06	0.04	0.04	Lr.

در جدول (۵)، در ست داده‌های 3-Spiral و Wisconsin دقت‌های قابل قبولی برای تمامی الگوریتم‌ها حاصل شده است و دلیل اصلی آن را می‌توان فاصله مناسب کلاس‌ها از یکدیگر دانست. در ست داده Banana به دلیل درهم‌رفتگی بین کلاس‌ها، الگوریتم‌ها توانایی تفکیک کامل

جدول ۳: میانگین دقت دسته‌بندی در ست داده Iris به ازای ۲۰ اجرای الگوریتم‌ها.

الگوریتم‌ها	دقت	زمان (S.)
NALM	96.67 ± 4.66	0.0317
ALM	95.33 ± 3.22	0.1043
ANFIS	95.33 ± 4.50	0.9332
MLP	96.67 ± 3.51	0.7847

همان‌طور که مشاهده می‌شود الگوریتم پیشنهادی با سرعت بیشتر و حجم محاسبات کمتر توانسته به دقتی مشابه دقت الگوریتم رایج ALM دست یابد. همان‌طور که از میزان انحراف معیار می‌توان نتیجه‌گیری کرد، این روش‌ها در مقایسه با روش شبکه‌های عصبی MLP دارای پایداری بیشتر و نوسان کمتری می‌باشند.

در شبیه‌سازی دیگر به مقایسه عملکرد الگوریتم‌های فوق در برخورد با داده‌های Aggregation پرداخته شده است و نتایج حاصل از این شبیه‌سازی‌ها در جدول (۴) آورده شده است. برای شبیه‌سازی‌ها در شبکه MLP تعداد ۲ نرون به‌عنوان ورودی و ۱ نرون به‌عنوان نرون خروجی و در لایه نهان ۲۲ نرون قرار گرفته شده است. تابع فعالیت نرون‌ها از نوع سیگموئید در نظر گرفته شده است و الگوریتم یادگیری نیز لونیگ-مارکار که مقدار Rate یادگیری برابر با ۰/۰۸ در نظر گرفته شده است. در شبکه ANFIS، از تابع genfis1 و با دو تابع عضویت دوزنقه‌ای استفاده شده است. تعداد ایپاک‌ها برای الگوریتم MLP برابر با ۱۰۰ و برای الگوریتم ANFIS برابر با ۶۰۰ است. جدول (۳) به مقایسه خروجی حاصل از الگوریتم‌های مختلف پرداخته شده است. نمایشی این ست داده در شکل (۹) آورده شده است. در روش پیشنهادی مقادیر پارامترهای  $\beta_2 = 0.65$ ،  $\beta_1 = 0.04$  و  $\sigma = 8$  و  $lr = 16$  (۶٪ رزولوشن صفحات) انتخاب شده‌اند، در روش ALM شعاع پخش قطره جوهر برابر با ۱۶ (۶٪ رزولوشن صفحات) است. تعداد افزایش ازای متغیرها به ترتیب برابر با ۵ و ۷ است.

بالسی می‌گردند. بنابراین می‌توان الگوریتم ارائه‌شده را به‌عنوان یک سیستم نرو فازی در نظر گرفت. وجود ضریب  $\alpha$  در الگوریتم پیشنهادی خاصیت وفقی را به آن داده است. با کاهش ضریب آلفا در مراحل جلوتر الگوریتم، میزان تأثیر داده‌های جدید بر خروجی کاهش یافته و میزان تأثیر داده‌های پیشین که نقش همان تجربه را در یادگیری انسانی دارند افزایش می‌یابد (مهم بودن ترتیب داده‌ها و اهمیت دانش قبلی). پیاده‌سازی سخت‌افزاری بر روی بستر ممریستور یا FPGA از کارهای آینده خواهد بود.

## مراجع

- [1] L. A. Zadeh, *Fuzzy sets*, Information and control, vol. 8, no. 3, pp. 338-353, 1965.
- [2] Tanaka, K. and H.O. Wang, *Fuzzy control systems design and analysis: a linear matrix inequality approach*, John Wiley & Sons, 2004.
- [۳] پوریا جعفری، محمد تشنه‌لب، مهسان توکلی کاخکی، طراحی کنترل‌کننده فازی تطبیقی مستقیم برای سیستم‌های مرتبه کسری غیرخطی به کمک جبران‌ساز، مجله مهندسی برق دانشگاه تبریز، جلد ۴۷، شماره ۳، ۱۳۹۶.
- [4] Mamdani, E.H., *Application of fuzzy algorithms for control of simple dynamic plant*, Electrical Engineers Proceedings of the Institution of, vol. 121, no. 12, pp. 1585-1588, 1974.
- [5] Takagi, T. and M. Sugeno, *Fuzzy identification of systems and its applications to modeling and control*, IEEE transactions on systems, man, and cybernetics, vol. 1, pp. 116-132, 1985.
- [6] Terano, T., K. Asai, and M. Sugeno, *Applied fuzzy systems*, Academic Press, 2014.
- [۷] ندا خانبانی، امیر مسعود افتخاری مقدم، ارائه یک روش تشخیص زبان علامت مبتنی بر رویکرد *MLRF* فازی با استفاده از اطلاعات عمق تصویر، مجله مهندسی برق دانشگاه تبریز، جلد ۴۷، شماره ۳، ۱۳۹۶.
- [۸] سمیرا رفیعی، پرهام مرادی، بهبود عملکرد الگوریتم خوشه‌بندی فازی سی-مینز با وزن‌دهی اتوماتیک و محلی ویژگی‌ها، مجله مهندسی برق دانشگاه تبریز، جلد ۴۶، شماره ۲، ۱۳۹۵.
- [9] Bezdek, J.C., *Fuzzy mathematics in pattern classification*, 1973.
- [10] Del Campo, I., et al, *Efficient hardware/software implementation of an adaptive neuro-fuzzy system*, IEEE Transactions on Fuzzy Systems, vol. 16, no. 3, pp. 761-778, 2008.
- [11] Himavathi, S., D. Anitha, and A. Muthuramalingam, *Feedforward neural network implementation in FPGA using layer multiplexing for effective resource utilization*, IEEE Transactions on Neural Networks, vol. 18, no. 3, pp. 880-888, 2007.
- [12] Soudry, D., et al., *Memristor-based multilayer neural networks with online gradient descent training*, IEEE transactions on neural networks and learning systems, vol. 26, no. 10, pp. 2408-2421, 2015.
- [13] Ortega-Zamorano, F., et al., *Efficient implementation of the backpropagation algorithm in fpgas and microcontrollers*, IEEE transactions on neural networks and learning systems, vol. 27, no. 9, pp. 1840-1850, 2016.
- [14] Shouraki, S.B., *A novel fuzzy approach to modeling and control and its hardware implementation based on brain functionality and specifications*, 2000.

کلاس‌ها را از یکدیگر ندارند. بنابراین همان‌طور که از نتایج جداول قابل مشاهده است الگوریتم پیشنهادی توانسته به‌خوبی عملکردی قابل توجه نسبت به سه الگوریتم دیگر داشته باشد و همچنین واریانس خروجی نسبت به سایر الگوریتم‌ها کمتر است که دلیل اصلی این موضوع در عدم نیازمندی الگوریتم‌ها به مقداردهی اولیه است. نکته دیگری که حائز اهمیت است، اینکه الگوریتم پیشنهادی در زمان کمتری نسبت به سایر الگوریتم‌ها می‌تواند به خروجی مطلوب دست یابد و نیازمند هیچ‌گونه پیش‌پردازش اولیه‌ای بر روی داده‌های آموزشی نیست و همین نشان از توانمندی بالای الگوریتم نسبت به الگوریتم‌های مشابه خود نظیر شبکه‌های عصبی مصنوعی دارد. سرعت الگوریتم ALM نسبت به الگوریتم MLP و ANFIS بیشتر بوده [۳۶] و البته این نتایج در جداول شبیه‌سازی نیز قابل مشاهده است. همچنین سرعت الگوریتم پیشنهادی بسیار بهتر از ALM است و دلیل اصلی آن حذف اپراتور پخش قطره جوهر و همچنین محاسبات مرتبط با یافتن ویژگی‌ها است. قابل‌بیان است که روش پیشنهادی همانند روش ALM برخلاف روش‌های شبکه عصبی که همانند جعبه سیاه عمل می‌کنند یک روش دانش‌محور بوده و بر اساس استخراج دانش موجود در فضای داده عمل می‌کند و همین سبب می‌شود که در برخورد با مسائل پیچیده همانند شبکه‌های عصبی مصنوعی نیازمند بهره‌مندی از پیش‌پردازش‌ها روی فضای داده نیست.

## ۵- نتیجه

یکی از ابزارهای ارائه شده بر پایه طبقه‌بندی رفتار مغز انسان ALM است. این الگوریتم علی‌رغم کارایی مناسب در انواع کاربردها از حجم محاسباتی زیاد مورد نیاز رنج می‌برد و سبب شده که کمتر مورد توجه قرار گیرد. در این مقاله یک الگوریتم یادگیری جدید بر مبنای سیستم محاسباتی IDS ارائه شده که با بهره‌گیری از بردارهای توصیفی برای صفحات IDS علاوه بر کارایی مناسب همانند الگوریتم پیشین، توانسته دانش موجود در زیرسیستم‌ها را با هزینه کمتر استخراج کرده و از دانش استخراج‌شده در لایه‌های دیگر استفاده کند. در این مقاله با استفاده از روش ارزیابی 10-fold validation، سرعت پردازش و پیچیدگی محاسباتی بررسی شده و افزایش چشم‌گیر سرعت الگوریتم پیشنهادی در مقایسه با الگوریتم‌های ANFIS، MLP به‌ازای ست‌داده‌های گوناگون در مسائل طبقه‌بندی و مدل‌سازی قابل مشاهده بوده و بنابراین در صورت پیاده‌سازی سخت‌افزاری، کاهش حجم سخت‌افزاری زیادی در پی خواهد داشت.

در الگوریتم ارائه‌شده داده‌های ورودی و خروجی به‌صورت فازی بوده و در مرحله یادگیری به‌ازای هر داده آموزشی به‌روزرسانی مقادیر بردارها همانند روش یادگیری SOM<sup>۱۹</sup> صورت می‌گیرد (مشابه رابطه ۱۴). همچنین با در نظر گرفتن هر سطح کوانتیزاسیون به‌عنوان یک نرون، به‌ازای هر ورودی، ضرایب مجموعه‌ای از نرون‌ها تغییر کرده و تکرار داده‌های ورودی سبب تقویت ضرایب همانند شبکه‌های عصبی

- [25] Bahrpeyma, F., A. Zakerolhoseini, and H. Haghghi, Using IDS fitted Q to develop a real-time adaptive controller for dynamic resource provisioning in Cloud's virtualized environment, *Applied Soft Computing* 26, pp. 285-298, 2015.
- [26] Klidbary, S.H., et al. Outlier Robust Fuzzy Active Learning Method (ALM), in 7th International Conference on Computer and Knowledge Engineering (ICCKE 2017), October 26-27 2017, Ferdowsi University of Mashhad. 2017.
- [27] Shouraki, S.B., N. Honda, and G. Yuasa, *Fuzzy interpretation of human intelligence*, International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, vol. 7, no. 4, pp. 407-414, 1999.
- [28] Firouzi, M., S.B. Shouraki, and I.E.P. Afrakoti, *Pattern analysis by active learning method classifier*, Journal of Intelligent & Fuzzy Systems, vol. 26, no. 1, pp. 49-62, 2014.
- [29] Javadian, M., S.B. Shouraki, and S.S. Kourabaslou, A novel density-based fuzzy clustering algorithm for low dimensional feature space, *Fuzzy Sets and Systems*, 2016.
- [30] Afrakoti, I.E.P., S.B. Shouraki, and B. Haghghat, *An Optimal Hardware Implementation for Active Learning Method Based on Memristor Crossbar Structures*, IEEE Systems Journal, vol. 8, no. 4, pp. 1190-1199, 2014.
- [31] Shouraki, S.B. and N. Honda, *Recursive Fuzzy Modeling Based on Fuzzy Interpolation*, JACIII, vol. 3, no. 2, pp. 114-125, 1999.
- [32] Firouzi, M. and S.B. Shouraki, *Performance Evaluation of Active Learning Method in Classification Problems*, 3rd International Conference on Machine Learning and Computing, ICMLC. 2011.
- [33] Bishop, C.M., *Pattern recognition*, Machine Learning, 128, 2006.
- [34] Rini, D.P., S.M. Shamsuddin, and S.S. Yuhaniz, *Particle swarm optimization for ANFIS interpretability and accuracy*, Soft Computing, vol. 20, no. 1, pp. 251-262, 2016.
- [35] B. Gliwa, and A. Byrski, *Hybrid neuro-fuzzy classifier based on NEFCLASS model*, Computer Science 12, pp. 115, 2013.
- [36] I. E. P. Afrakoti, et al., Using a memristor crossbar structure to implement a novel adaptive real-time fuzzy modeling algorithm, *Fuzzy Sets and Systems*, 307, pp.115-128, 2016.
- [15] Sugeno, M. and T. Yasukawa, *A fuzzy-logic-based approach to qualitative modeling*, IEEE Transactions on fuzzy systems, vol. 1, no. 1, pp. 7-31, 1993.
- [16] Murakami, M. and N. Honda, A study on the modeling ability of the IDS method: A soft computing technique using pattern-based information processing, *International journal of approximate reasoning*, vol. 45, no. 3, pp. 470-487, 2007.
- [17] Sakurai, Y., A study of the learning control method using PBALM-a nonlinear modeling method, PhD, The University of Electro-Communications, Tokyo, 2005.
- [18] Shouraki, S.B. and N. Honda. Fuzzy controller design by an active learning method, in 31th Symposium of Intelligent Control. Tokyo, Japan. 1998.
- [19] Ghorbani, M.J., M.A. Choudhry, and A. Feliachi. Distributed multi-agent based load shedding in power distribution systems, in Electrical and Computer Engineering (CCECE), 2014 IEEE 27th Canadian Conference on. 2014.
- [20] MURAKAMI, M., Practicality of modeling systems using the IDS method: Performance investigation and hardware implementation, The University of Electro-Communications, 2008.
- [21] Firouzi, M., S.B. Shouraki, and J. Conrardt. Sensorimotor Control Learning Using a New Adaptive Spiking Neuro-Fuzzy Machine, Spike-IDS and STDP, in International Conference on Artificial Neural Networks, Springer, 2014.
- [22] Cranganu, C. and F. Bahrpeyma, Use of active learning method to determine the presence and estimate the magnitude of abnormally pressured fluid zones, a case study from the Anadarko Basin, Oklahoma, in Artificial Intelligent Approaches in Petroleum Geosciences, Springer. pp. 191-20, 2015.
- [23] Bahrpeyma, F., C. Cranganu and B.Z. Dadaneh, Active learning method for estimating missing logs in hydrocarbon reservoirs, in Artificial Intelligent Approaches in Petroleum Geosciences, Springer. pp. 209-224, 2015.
- [24] Merikh-Bayat, F., F. Merikh-Bayat, and S.B. Shouraki, The neuro-fuzzy computing system with the capacity of implementation on a memristor crossbar and optimization-free hardware training, *IEEE Transactions on Fuzzy Systems*, vol. 22, no. 5, pp. 1272-1287, 2014.

<sup>18</sup> Not Linearly Separate able<sup>19</sup> Self-Organizing Map (SOM)

زیر نویس ها

- <sup>1</sup> Active Learning Method (ALM)
- <sup>2</sup> Ink Drop Spread (IDS)
- <sup>3</sup> Multi-Layer Perceptron (MLP)
- <sup>4</sup> Adaptive Neuro-Fuzzy Inference System (ANFIS)
- <sup>5</sup> Fuzzy Inference
- <sup>6</sup> State of the art
- <sup>7</sup> Error Back Propagation (EBP)
- <sup>8</sup> Memristor
- <sup>9</sup> Locally
- <sup>10</sup> Create
- <sup>11</sup> Fraction of Variance Unexplained (FVU)
- <sup>12</sup> Correlation
- <sup>13</sup> Levenberg-Marquardt
- <sup>14</sup> Epochs
- <sup>15</sup> Batch-Mode
- <sup>16</sup> Sample-Mode
- <sup>17</sup> O-notation